

Listing of the Claims:

1-19. (Canceled)

20. (Currently Amended) A method of generating test data to test an application, the method comprising:

~~identifying a data field and a data value for the data field to test an application;~~
~~defining in a property file a set of data field identifiers with a data value corresponding to each of the data field identifiers;~~
~~providing a data definition including the data field and the data value to a property file;~~
~~defining in the property file at least one test string format as a sequence of a subset of the data field identifiers in the property file,~~
~~wherein the definitions in the property file do not adhere to a COBOL copybook syntax;~~
providing a[[n]] command prompt instruction for managing generating a test file;
~~generating, based on an instruction, a string of test data the data field and data value to the test file using based on the instruction, wherein the string of test data is structurally equivalent to a data string output by a COBOL application and includes a sequence of the data values that are not delimited by any special character and arranged in accordance with the sequence of corresponding field identifiers defined by one of the test string formats in the property file, the test file used to test the application;~~
testing the application with the test file; and

modifying the contents of the test file with a change test file command prompt instruction.

21-27. (Canceled)

28. (New) The method of claim 20, further comprising:

defining in the property file an indication of which of the at least one test string formats are to be used to generate test files.

29. (New) The method of claim 28, wherein a separate test file is generated for each of the test string formats in the indication.

30. (New) The method of claim 20, further comprising:

defining in the property file a second set of data field identifiers without a corresponding data value, wherein the test string format is defined in the property file to also include a subset of the second set of data field identifiers in the sequence.

31. (New) The method of claim 30, wherein a null value is generated in the string of test data for each data field identifier that belongs to the second set of field identifiers.

32. (New) The method of claim 30, wherein the command prompt instruction includes an argument defining a data value for at least one data field identifier that belongs to the second set of field identifiers to be used in the string of test data.

33. (New) The method of claim 20, wherein the command prompt instruction includes an argument defining a data value for at least one data field identifier to be used in the string of test data instead of the data value defined in the property file.

34. (New) The method of claim 20, wherein each field identifier is a COBOL data definition.

35. (New) The method of claim 20, further comprising:

editing at least one data value in the property file; and

generating a new string of test data in accordance with the edited property file.

36. (New) The method of claim 35, wherein the new string of test data is generated to overwrite the test file.

37. (New) The method of claim 35, wherein the new string of test data is generated to a new test file.

38. (New) A method of generating test data to test an application, the method comprising:
- defining in a property file a plurality of data field identifiers with a data value corresponding to at least one of the data field identifiers, and at least one test string format as a sequence of the data field identifiers, wherein the definitions in the property file do not adhere to a COBOL copybook syntax;
- generating a first test file corresponding to the test string format based on a new test file command prompt instruction, wherein the first test file is created to include a string of the data values that is structurally equivalent to a data string output by a COBOL application and the data values are not delimited by any special character and are arranged in accordance with the sequence of corresponding field identifiers in one of the test string formats;
- testing the application with the test file; and
- modifying the contents of the first test file with a change test file command prompt instruction.

39. (New) The method of claim 38, wherein a null value is generated in the string of test data for each data field identifier that does not have a corresponding data value.

40. (New) The method of claim 38, wherein the new test file command prompt instruction includes an argument defining a data value for at least one data field identifier to be used in the string of test data instead of the data value defined in the property file.

41. (New) The method of claim 38, wherein the change test file command prompt instruction includes an argument defining a new data value for at least one data field identifier, wherein each of the data values in the string corresponding to the data field identifier in the argument are overwritten in the first test file with the new data value.

42. (New) The method of claim 41, wherein the data values in the string not corresponding to the data field identifier in the arguments are unchanged in the first test file.

43. (New) The method of claim 38, further comprising:

producing an empty property file that includes the data field identifiers and the at least one test string format in the property file; and

editing the empty property file to associate a data value with each data field identifier that is to be changed in the first test file, wherein the contents of the first test file are modified by each of the data values in the string corresponding to the data field identifiers that have an associated data value in the edited property file being overwritten in the first test file with the corresponding data value in the edited property file.

44. (New) The method of claim 38, further comprising:

producing an empty property file that includes the data field identifiers and the at least one test string format in the property file, wherein the change test file command prompt instruction includes arguments defining new data values for each data field identifier that is to be

changed, wherein each of the data values in the string corresponding to the data field identifiers in the argument are overwritten in the first test file with the corresponding new data value.

45. (New) The method of claim 44, wherein only data values in the string in the first file corresponding to data field identifiers in the arguments are overwritten.

46. (New) The method of claim 38, further comprising:

editing at least one test string format in the property file.